

Self-supervised Social Relation Representation for Human Group Detection

Jiacheng Li^{1*}, Ruize Han^{1*†}, Haomin Yan¹, Zekun Qian¹,
Wei Feng¹, and Song Wang²

¹ Intelligence and Computing College, Tianjin University, China

² University of South Carolina, Columbia, USA

{threeswords, han_ruize, yan_hm, clarkqian, wfeng}@tju.edu.cn,
songwang@cec.sc.edu

Abstract. Human group detection, which splits crowd of people into groups, is an important step for video-based human social activity analysis. The core of human group detection is the human social relation representation and division. In this paper, we propose a new two-stage multi-head framework for human group detection. In the first stage, we propose a human behavior simulator head to learn the social relation feature embedding, which is self-supervised trained by leveraging the socially grounded multi-person behavior relationship. In the second stage, based on the social relation embedding, we develop a self-attention inspired network for human group detection. Remarkable performance on two state-of-the-art large-scale benchmarks, i.e., PANDA and JRDB-Group, verifies the effectiveness of the proposed framework. Benefiting from the self-supervised social relation embedding, our method can provide promising results with very few (labeled) training data. We have released the source code to the public.

Keywords: Group detection, self-supervised learning, video analysis

1 Introduction

Scenes with a crowd are very common in the real world, e.g., outdoor parades and social gatherings [20,21]. Such scenes lead to many important video-surveillance tasks, such as human counting, pedestrian tracking [19,18,13], and group activity recognition [45]. These tasks involve crowd analysis either locally for individuals or more globally for the whole crowd. With the use of wide-view cameras in surveillance, we can now collect videos to cover a large-scale crowd composed of multiple groups of people with high resolution. In this case, we usually need to first identify all these groups [27] before conducting further human activity analysis. This leads to the important task of (video-based) human group detection, which tries to divide the crowd into multiple non-overlapped human groups.

The human group detection task is very challenging since groups can be formed in complicated ways [22,34], whose characteristic is more complex than

*Equal Contribution. †Corresponding Author.

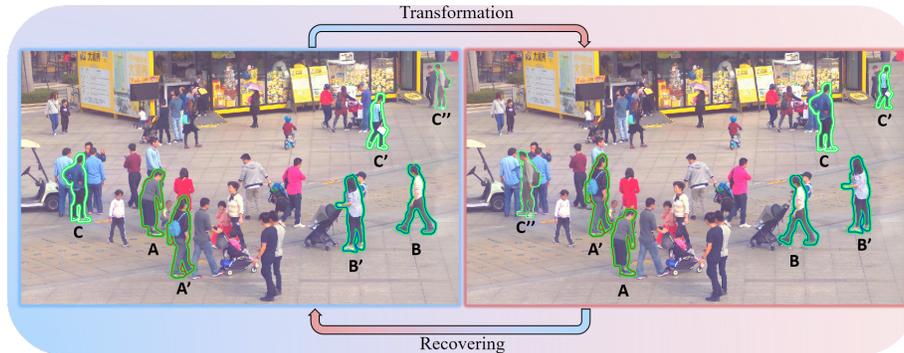


Fig. 1. A conceptual illustration of the rationale of our insight. We swap some people in the scene and try to discover the unreasonable states in the transformed scene (right) by implicitly learning the human social relations. Specifically, in the left image, we swap person A with A' , B with B' and C, C', C'' with C', C'', C , respectively, resulting in the right image. Such swapped results in the right contain persons with abnormal behaviors. To recover the above augmentation, the method needs to understand the reasonable state of the person in the crowd considering itself and its surrounding people, which can be used for the social relation representation.

the pedestrians acting alone or the crowds as a whole. Previous works for group detection mainly leverage the features extracted from either the pedestrian trajectories [29] or local spatial structure [6], followed by a similarity measurement and clustering analysis, e.g., the weighted graph model [4], potentially infinite mixture model [15] and correlation clustering [34]. In a recent work of PANDA [39], a new video benchmark that uses gigapixel-level cameras for capturing a supersized scene with hundreds of people was developed, together with an official baseline method that combines the global trajectory and local interaction information for group detection.

However, the above methods commonly have two weaknesses. First, the features used in most of the previous group detection methods are pre-defined, many are actually hand-crafted features, including temporal trajectories and spatial distances [30,41,6,34]. Second, previous works based on machine learning are usually fully supervised and require large-scale annotations [41,39]. Nevertheless, the human group detection labels in a dense crowd are very laborious and costly to annotate manually.

To address these problems, in this paper, we study spontaneous learning of human relation embedding features for human group detection. This is inspired by the sociological interpretation of group – two or more people interacting to reach a common goal and perceiving a shared membership, based on both physical identity (spatial proximity) and social identity (intra-group rules) [37]. For a person in the crowd scene, her/his activity and motion characteristics are highly influenced by the nearby people, especially the interactive ones. For example, as shown in Figure 1, we swap the spatial positions of some people (left and

right images are the ones before and after the swapping). We can easily find that the ‘transformed’ people in the right image show unreasonable behaviors, e.g., in the right of Figure 1, the strange pose (A, B’), dangerous moving direction with close distance to something (A’, B). Such visual unreasonability, in the human sense, is dependent on the human-human interactive behavior and social relation, which are very important for human group detection. Inspired by this rationale, we try to develop a method to automatically learn human social relations through such unreasonable behaviors. More specifically, we design a rational-behavior simulator head (BS-Head) to discover the swapped persons and recover their original states based on the information of the surrounding persons. This head is used as a self-supervised training head to learn the human social relation embedding. After that, we use the embedding feature produced by BS-Head to train another self-attention-based group prediction network for human group detection. The main contributions in this work are:

- We propose a two-branch group detection framework, which contains a shared relation embedding module and two heads for two-stage training. It is effective for social group detection problems in large-scale crowds.
- We develop a self-supervised relation embedding training strategy, which models the socially grounded human social relations in the crowd scene as the pretext task. The self-supervised representation can benefit the network training using only very few (labeled) data.
- Experiment results on two newly proposed large-scale benchmarks, i.e., PANDA and JRDB-Group, verify the effectiveness of the proposed method. The proposed method achieves significant group detection performance improvement compared to the state-of-the-art baselines – $20.7 \rightarrow 53.2\%$ and $31.4\% \rightarrow 56.9\%$ in F_1 score on PANDA and JRDB-Group, respectively. We release the source code to the public at <https://github.com/Jiaoma/SHGD>.

2 Related Work

Human group detection. Human group detection is an essential task in computer vision, which, however, has not been widely studied recently. Early human group detection methods can be divided into three categories, including group-based methods, individual-group methods, and individual-based methods. In group-based methods, no individual information is considered [33,11]. The individual-group methods try to integrate the information of human trajectories instead of separately using each one [28,3]. The individual-based methods consider individual subjects, leading to many models, such as predicting the social group relations by using the weighted graph [4], potentially infinite mixture model [15] and correlation clustering [34], etc. Recently, Shao et al. [32] employed the goal directions instead of traditional positions and velocities to find group members. More recently, instead of focusing only on the position-aware group detection, PANDA [39] proposes to integrate the dynamic human interactions for human group detection. As mentioned earlier, these methods either

use pre-defined, usually hand-crafted, features [30,41,6,34] or require large-scale annotated data [41,39] for training.

Human relation discovery. A key problem for the group detection task is the discovery of human relations, especially social relationship [23,16], e.g., friends and colleagues, via various human attributes, e.g., age, job, etc. Other relations include the human-object interaction (HOI) [46,47], human-human interaction (HHI) [26,45], and gaze communication [10], etc. For HOI or HHI, relations are usually specific and mostly determined by the involved subjects/objects, e.g., a scene with a human and a bike is more likely to be detected as ‘ride bike’. Differently, our task explores the relations without the above specific prior knowledge, e.g., the social relationships among people in the crowd.

Human relation modeling in the multi-person scene. Group-wise human relation is also collaboratively or implicitly used in many other related tasks, and human group detection is often accompanied by pedestrian trajectory prediction [30,41,12]. The method in [30] tries to jointly achieve the pedestrian trajectory prediction and group membership estimation using a Conditional Random Field (CRF) model. Similarly, an SVM-based method is proposed in [41] to handle the estimations of pedestrian trajectory and social (group) relationships. A GAN pipeline is proposed in [32] that learns informative latent features for joint pedestrian trajectory forecasting and group detection. Social-GAN [17] proposes to predict the socially-accepted motion trajectories in crowded scenes by considering human social relations. Multi-human relations have also been considered in group activity recognition (GAR) [40,42,2,14,31,43]. For example, ARG [40] proposes to build a flexible and efficient actor relation graph to simultaneously capture the appearance and position relation between actors. The method in [31] considers multiple cliques with different scales of the locality to account for the diversity of the actors’ relations in group activities. The above methods all take the human relation modeling or detection as an auxiliary task but do not divide the crowd into groups that is our focus in this paper.

3 Proposed Method

3.1 Overview

We first give an overview of the proposed method. Given a video clip recording a crowded multi-person scene, we denote N as the maximum number of people (referred to as subject in this paper) within T frames¹, e.g., 10 frames, where N could be very large, e.g., over 1,000 in PANDA dataset. We first extract a feature encoding vector $\mathbf{v}_i^t \in \mathbb{R}^{C_v}$ for the i -th subject in t -th frame, where C_v is the number of feature dimensions. The encoding vector can be represented by both the human skeleton joints and spatial location. We model all the the subjects in the scene as an undirected graph $\mathcal{G} = \langle \mathbf{V}, \mathbf{R} \rangle$, in which $\mathbf{V} \in \mathbb{R}^{T \times N \times C_v}$ denotes the set for all \mathbf{v}_i^t , and $\mathbf{R} \in \mathbb{R}^{T \times N^2}$ encodes the *group relation* $r_{i,j}^t$ between subject i and subject j at frame t . It is in the range of $[0, 1]$, where 1 means the subject

¹ If a subject is missing in a frame, we fill it with blank (all-zero feature vector).

j and subject i belong to the same group. The relation matrix \mathbf{R} can be used to divide the crowd of people into groups by label propagation methods [44].

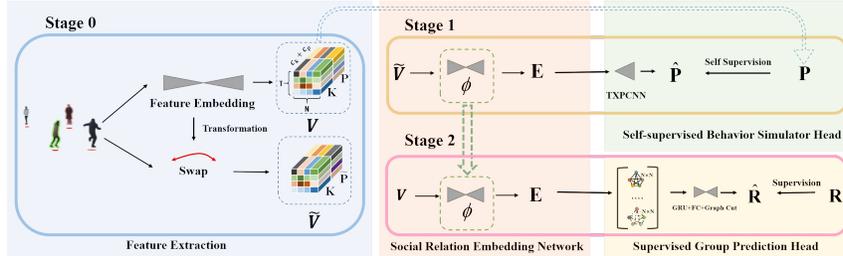


Fig. 2. An illustration of the overall framework of our method, which is composed of an offline feature extraction module and a social relation embedding module for feature construction, followed by two heads, i.e., the behavior simulator head (BS-Head) and the group prediction head. The overall network is trained in two stages. In stage 1, we train the relation embedding network with the BS-Head in a self-supervised manner. In stage 2, based on the relation embedding features obtained by stage 1, we train our network with the group prediction head for generating the group detection results.

To obtain such relation matrix \mathbf{R} , many deep learning methods [36,35] generally first encode the features \mathbf{V} into a group relation embedding space and then predict \mathbf{R} using a relation modeling algorithm, like the graph neural networks (GNN). In this paper, we aim at developing an effective group relation embedding method in a self-supervised manner, and then use such embedding \mathbf{E} to predict the relation matrix \mathbf{R} . The pipeline of the proposed method could be summarized as two stages: 1) We first train a real-world surrounding-subject-aware *human behavior simulator* in a self-supervised manner, which is to spontaneously learn the natural human behavior within a group and embed the features into a relation embedding space. 2) Based on the relation embedding, we then learn a group prediction sub-network, which is trained for discovering the human-human group relations among multiple persons. This stage can only use a small number of labeled data to train. The architecture of the whole network is shown in Figure 2. It consists of two prediction heads that share the same feature extraction and social relation embedding network. The whole network uses the human detection bounding boxes of the N persons each frame in T frames as input for feature construction, as discussed in Section 3.2. It has two kinds of outputs produced by its two heads separately in the two-stage training, as presented in Section 3.3 and Section 3.4, respectively.

3.2 Feature Construction Module

Social relation embedding network. The original feature $\mathbf{V} \in \mathbb{R}^{T \times N \times C_v}$ is constructed by concatenating both the 2D skeleton joints $\mathbf{K} \in \mathbb{R}^{T \times N \times C_k}$ and the positional feature $\mathbf{P} \in \mathbb{R}^{T \times N \times C_p}$ of all N subjects at frame t .

The social relation embedding network uses \mathbf{V} as input and outputs a social relation embedding feature \mathbf{E} , i.e.,

$$\mathbf{E} = \phi(\mathbf{V}) \in \mathbb{R}^{T \times N \times C_e}. \quad (1)$$

Specifically, first, for the skeleton feature \mathbf{K} , we adopt the Shift-GCN [5] alike network to extract the temporal action features from the human skeleton joint. Also, for the positional feature \mathbf{P} , we concatenate it with the skeleton feature \mathbf{K} and utilize the architecture in spatial-temporal graph convolutional network (ST-GCNN) [25] to model the spatial and temporal information.

3.3 Self-supervised Behavior Simulator Head

In this section, we present the rationale and process to train the social relation embedding network in a self-supervised manner with the proposed rational-behavior simulator head (BS-Head). Specifically, we aim to establish a human behavior simulator to model the *reasonable behavior* of the humans in a crowd with the constraint of its social relation/rules/etiquette with other people, e.g., the greeting from surrounding people and the relative position distribution to the others. Our basic assumption is that every person should behave logically and not do unreasonable things like talking to nobody, stepping over others, totally ignoring others' greetings, etc. This way, we assume that a person with its surrounding subjects can be modeled by an implicit distribution D . Specifically, given a person c and its neighbors (with closer distance), we have

$$H_c \triangleq (\mathbf{V}_c, \mathbf{V}_n) \sim D, n \in \mathcal{N}_c, \quad (2)$$

where \mathbf{V}_c is the extracted feature of the center subject c , and \mathbf{V}_n with $n \in \mathcal{N}_c$ denotes the features of the neighboring subjects, which are identified by the spatial distance (in the image) to c within a threshold. We define H_c as the *neighbor feature cluster* by pairing the center subject and surrounding subjects.

To learn such distribution, we propose a self-supervised learning method. Our basic idea is that if the state (e.g., spatial position, human behavior) of a center subject is artificially changed, the states of its surrounding subjects can synergistically guide the center subject to be recovered to its original state. This way, as shown in Figure 3, we first destroy the state of the feature block \mathbf{V} and vary it by spatial-temporal transformation, i.e., swapping its position with another one, as shown in Figure 1. We can perform the swapping directly in the feature space, to get $\tilde{\mathbf{V}}$ as

$$\tilde{\mathbf{V}} = \chi(\mathbf{V}) \in \mathbb{R}^{N \times C_v}, \quad (3)$$

where χ denotes the transformation operation.

The transformed feature $\tilde{\mathbf{V}}$ and its surrounding subjects are then fed to the BS-Head, which tries to recover the original feature \mathbf{V} . This way, with the BS-Head, the training process of the first-stage network can be implemented as

$$\hat{\mathbf{V}} = \text{BSHead}(\phi(\tilde{\mathbf{V}})) \rightarrow \mathbf{V}, \quad (4)$$

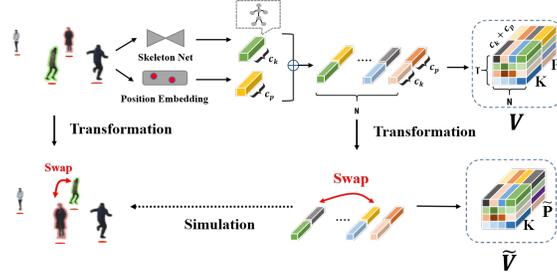


Fig. 3. An illustration of the feature transformation.

where BSHead denotes the rational-behavior simulator head. It tries to generate a recovered feature $\tilde{\mathbf{V}}$ tending to approximate the original feature \mathbf{V} . Here ϕ denotes the social relation embedding network as discussed in Section 3.2. For the behavior simulator head (BS-Head), we adopt the structure in the graph convolution neural network TXP-CNN [25].

Discussion. As shown in the left of Figure 4, the transformed feature $\tilde{\mathbf{V}} \in \mathbb{R}^{T \times N \times C_v}$ is constructed by the skeleton feature \mathbf{K} and the positional feature $\hat{\mathbf{P}}$, where the transformation operation χ is only applied to the positional feature \mathbf{P} , i.e., we simulate to exchange the position of a subject with another while maintaining its skeleton (representing the action and pose of a human). This may make the subject with original behavior in the swapped position look unreasonable with its (new) neighbors (see subjects A and B' in the right of Figure 1). Also, the BSHead actually only outputs the recovered positional feature $\hat{\mathbf{P}}$ (not including the skeleton feature), which is self-supervised by that from the original feature block \mathbf{P} , as shown in Figure 2.

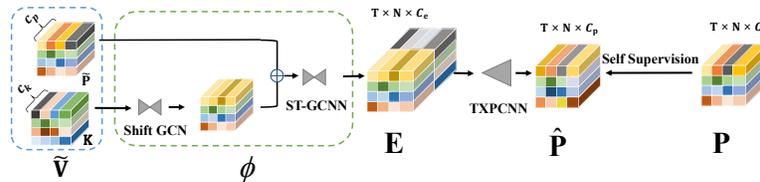


Fig. 4. An illustration of the self-supervised behavior simulator head, which trains the relation embedding network with the BS-Head in a self-supervised manner.

3.4 Supervised Group Prediction Head

The above behavior simulator head is mainly used for training the social relation embedding network ϕ in a self-supervised manner, which, however, can not

provide the group detection results. We propose the group detection head, as shown in Figure 5, which consists of a stacked attention module (Stack-Att), a GRU, and a linear layer to predict the human relation matrix, which will be post-processed by a label propagation method [44] to generate the predicted human group results.

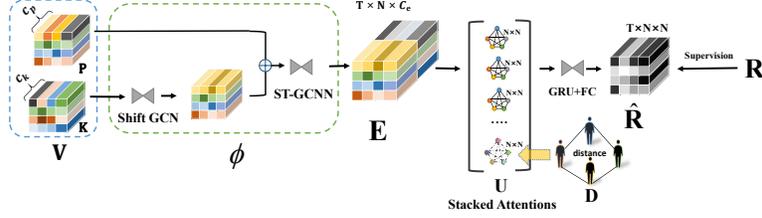


Fig. 5. An illustration of the second stage of our method, based on the relation embedding features obtained by the first stage, which trains the network with the group prediction head for generating the final group detection results.

This head takes the original feature block \mathbf{V} as input, which is first fed into the embedding network ϕ (trained by the first stage) to get social relation embedding $\mathbf{E} \in \mathbb{R}^{T \times N \times C_e}$. We then apply a stacked attention modules (Stack-Att), which uses M (a pre-set parameter) independent attention module operating separately as

$$\text{att}^m(\mathbf{e}_i, \mathbf{e}_j) = \frac{\theta(\mathbf{e}_i)^T \theta(\mathbf{e}_j)}{\sqrt{L}} \in \mathbb{R}, \quad m = 1, 2, \dots, M, \quad (5)$$

where $\mathbf{e}_i \in \mathbb{R}^{C_e}$ denotes an element of \mathbf{E} , and $\theta(\mathbf{e}_i) = \mathbf{W}_\theta \mathbf{e}_i + \mathbf{b}_\theta$ is the learnable linear transformations, L denotes the vector length of $\theta(\mathbf{e}_i)$. For each frame containing N subjects, we generate $N \times N$ pairs of elements (i, j) . For all $T \times N \times N$ element pairs (i, j) in T frames, we get the pair-wise relation map $\mathbf{A}^m \in \mathbb{R}^{T \times N \times N}$ with the value $A_{(i,j)}^m = \text{att}^m(\mathbf{e}_i, \mathbf{e}_j)$, calculated by the attention module. Then we stack all the relation map from M attention modules for all T frames to be a *stacked attention block* $\mathbf{U} \in \mathbb{R}^{T \times N \times N \times M}$ as a multi-channel relation feature block. Besides, considering the camera view in this problem is often very wide, and the persons with far distance couldn't be in one group, we also stack a distance matrix $\mathbf{D} \in \mathbb{R}^{T \times N \times N \times 1}$ on \mathbf{U} to get the final relation block $\mathbf{U}_D \in \mathbb{R}^{T \times N \times N \times (M+1)}$. This block is then fed to a single layer GRU temporal network to aggregate temporal information and a fully-connected layer (FC) to predict the final group relation matrix $\hat{\mathbf{R}} \in \mathbb{R}^{T \times N \times N}$, which is supervised by the annotated relation matrix \mathbf{R} , as shown in Figure 5.

Discussion. Inspired by the self-attention mechanism in [38] [40], the proposed Stack-Att adopts the self-attention for each pair of embedding features, i.e., $\mathbf{e}_i, \mathbf{e}_j$ to calculate their relation representation. Compared to the previous methods for relation (edge) feature representation, e.g., directly concatenating

the embedding features, the proposed strategy is more memory-efficient, which can handle the challenge of a large number of people in the crowd scene. Besides, the self-attention mechanism can better represent the relation between two embedding features. Our method is also different from [38] [40], which only produce 1-channel edge attention matrix. We apply it M times independently and stack them as a multi-channel edge feature block.

3.5 Implementation Details

Input features. As mentioned in Section 3.2, we use 2D skeleton joints \mathbf{K} and positional feature \mathbf{P} as input. We use Unipose network trained on MPII [1] dataset to detect the 2D skeleton joints of all the people in each frame. The detected 2D skeleton joints $\mathbf{k}_i^t \in \mathbb{R}^{32}$, for $i = 1, 2, \dots, N, t = 1, 2, \dots, T$ has 16 joints, each of which is a 2D coordinate in the pixel coordinate system. We also use the positional encoding method [38] to project the 2D position of each subject’s center to the 32-dimension space as the positional feature \mathbf{P} .

Two-stage training. In the first stage, we use the self-supervised training strategy as discussed in Section 3.3 to train the whole network using the relation-behavior simulator head, which outputs $\hat{\mathbf{P}}$ and we use MSE as loss criterion to enforce $\hat{\mathbf{P}}$ be close to \mathbf{P} . In the first training stage, we apply the swap operation to the training data. For the PANDA dataset, we first randomly select 10% persons in the scene and shuffle their positions as the swap operation. For JRDB dataset, since some frames only contain less than 10 persons, the selection ratio is 20%. Besides, to make the recovering of swap not equal to copying one’s origin position, we add small random noises following the uniform distribution on the swapped positions.

In the second stage, we use the trained social relation embedding network ϕ (in the first stage) with only the group prediction head to predict the group relation matrix $\hat{\mathbf{R}}$, which is enforced to be close to the ground-truth \mathbf{R} . Here we flatten the group relation matrix and use the cosine similarity loss as a criterion during training. This stage of training is a supervised training process, but it mainly updates the parameters of the group prediction head. This stage also fine-tunes the parameters of the social relation embedding network ϕ , which could further boost the performance of the whole framework on the group detection task. With the first stage self-supervised training for ϕ , the second stage can use few labeled data for training, which is also verified in the experiments.

We use stochastic gradient descent with Adam optimizer to optimize the parameters. The training of stage 1 takes 200 epochs with the learning rate of 1×10^{-4} . And the training of stage 2 takes 100 epochs with the learning rate of 7×10^{-3} . The proposed method is implemented based on the PyTorch framework.

Network inference. Given the input features \mathbf{V} , the inference stage only goes through the group prediction head and outputs the group relation matrix $\hat{\mathbf{R}} \in \mathbb{R}^{T \times N \times N}$. Then we use the label propagation method [44] to split N persons of each frame into non-overlapped groups, where the number of groups is automatically estimated by the algorithm. The groups only containing one person are dropped.

4 Experiments

4.1 Datasets and Metrics

- *PANDA benchmark*. We first choose the state-of-the-art dataset PANDA [39] for group detection evaluation. It is a multi-human video dataset covering real-world street scenes with a very wide field-of-view (1 km² area) and a very dense crowd in the scene, e.g., 4k subjects in one frame. For the group detection task, it provides 9 long-term videos in different scenes. The training set has 8 videos while the testing set has 1 video, as specified in the benchmark. The average number of frames and bounding boxes per video in the training set is 2,713 and 1,070.4k, respectively, while the number of frames and bounding boxes in the testing set is 3,500 and 335.2k, respectively. The average number of the human group per video is 144.6 and 75 in the training and testing video sets, respectively. The PANDA dataset provides human tracking for the group detection task and their baseline methods with evaluation metrics for comparison.

- *JRDB-Group benchmark*. We also include a new benchmark, i.e., JRDB-Group [8], for performance evaluation. JRDB-Group is built based on the JRDB dataset [24], which is captured by a panoramic camera equipped on a robot walking around in the crowded outdoor/indoor multi-person scenes. JRDB-Group provides the human social group annotation, and we use 20 training and 7 testing videos in JRDB-Group in our experiment. Following the setting in JRDB-Group, we evaluate the group detection task on the key frames, which are also sampled every 15 frames, generating 1,419 training and 404 testing samples.

Following the metrics in PANDA benchmark [39], we use the classical Half metrics [6] including precision, recall, and F₁ scores with group member *IoU* > 0.5 for group detection evaluation.

4.2 Results

Comparison methods. We find that most group detection methods are obsolete without available source code. We try our best to include more related methods with necessary modifications for comparison.

- *Dis.Mat* + [44]: We first consider a straightforward method in which we calculate the distance among all subjects in the crowd and apply the label propagation algorithm [44] used in our inference stage to get the group division.
- *GNN w GRU*: We apply a graph neural network (GNN) with the features in our method as the node feature to model the group relations among the subjects, in which we also apply a GRU model to integrate the temporal information.
- *ARG* is a state-of-the-art method [40] for human group activity recognition. ARG trains an Inception-v3 network to extract the appearance features and uses GNN to model the mutual relations among the subjects in the crowd. We use the affinity matrix in GNN as the relation matrix for group membership division.
- *Global-to-local* [39] is the baseline method proposed in PANDA [39], which applies a global-to-local zoom in framework to validate the incremental effectiveness of local visual clues to global trajectories. More specifically, human entities and

their relationships are represented as a graph. Then a global-to-local strategy is applied, and the video interaction scores among subjects are estimated by a spatial-temporal ConvNet. The edges in the graph are merged using label propagation, and the cliques remaining in the graph are the group detection results. • *Joint* and *JRDB-Group* are the baselines in JRDB-Group benchmark [8]. *Joint* [7] proposes to integrate the human group detection task into the group activity recognition problem. Based on it, *JRDB-Group* adds the human spatial features and number of group constraints as losses for the group detection task.

We compare the group detection results of the proposed method and other comparison methods. Table 1 shows the comparative results on the PANDA benchmark. We can see that all the comparison methods, including the state-of-the-art baseline method Global-to-local reported in the PANDA benchmark and its variants, generate unsatisfactory results. The comparison method with the best performance, i.e., *ARG*, generates an F_1 score of 25.4%. The proposed method significantly improves the group detection performance and provides a promising result with an F_1 score of 53.2%. We can see similar results on JRDB-Group as shown in Table 2, in which we can see that the comparison methods can generate relatively better results than in PANDA. We also surprisingly find that the simple baseline methods, i.e., the ‘distance matrix + [44]’ and the ‘GNN + GRU’, provide not bad performance. This is because the number of pedestrians and crowd density in JRDB-Group is lower than those in PANDA. The proposed method provides the best performance on JRDB-Group, which also outperforms the comparison methods by a large margin.

Table 1. Comparative group detection results on PANDA (%).

Method	Precision	Recall	F1
Dis.Mat + [44]	42.9	12.0	18.8
GNN w GRU	41.9	17.3	24.5
ARG [40]	34.9	20.0	25.4
Group-to-local [39]	23.7	12.0	16.0
Group-to-local w Random [39]	24.4	13.3	17.2
Group-to-local w Uncertainty [39]	29.3	16.0	20.7
Ours	55.9	50.7	53.2

4.3 Ablation Study

To verify the effectiveness of our method, we conduct the ablation study as below.

• **Ours w/o self-sup. 1st-train.** denotes removing the self-supervised representation training in the first stage in our method, and we directly train the whole network (including the representation embedding module and the group forming module) using the training dataset.

Table 2. Comparative group detection results on JRDB-Group (%).

Method	Precision	Recall	F1
Dis.Mat + [44]	57.3	23.5	33.4
GNN w GRU	43.4	28.6	34.5
ARG [40]	32.5	38.4	35.2
Joint [7]	30.0	28.4	29.1
JRDB-Group [9]	39.0	37.9	38.4
Ours	57.7	56.2	56.9

• **Ours w/o self-rep. fine-tune.** denotes removing the representation’s fine-tuning in our method, and we fix the parameters in the representation embedding module obtained by the self-supervised first-stage training and only train the group prediction head with the training dataset.

As shown in Table 3, we can first see that the proposed network with direct one-stage training can provide acceptable performance at both datasets, which outperforms the baseline methods significantly with F_1 scores of 44.6% on PANDA and 48.8% on JRDB-Group, respectively. Compared with the full two-stage training version of our method, we can also see that the self-supervised representation in the first-stage training is very effective, which further improves the F_1 score to 53.2% on PANDA and 56.9% on JRDB-Group, respectively. Besides, we find that representation network fine-tuning with the training dataset is also useful in our method.

Table 3. Ablation study results on PANDA and JRDB-Group datasets (%).

Method	PANDA			JRDB-Group		
	Precision	Recall	F1	Precision	Recall	F1
w/o (self-sup.) 1 st -train	48.4	41.3	44.6	49.8	47.8	48.8
w/o (self-rep.) fine-tune.	58.6	45.3	51.1	50.3	48.1	49.2
Ours	55.9	50.7	53.2	57.7	56.2	56.9

4.4 In-depth Analysis

Label annotations for the human group detection in a large-scale scene are laborious. Therefore, we propose the self-supervised relation representation, and further investigate the its performance using different amounts of training data.

As shown in Table 4, we can first see that the proposed method without the self-supervised first-stage training provides a poor performance with the reduction of training data. Specifically on the PANDA dataset, using very few training data, i.e., 10% of all, the group detection F_1 score is only 33.8%, which is much inferior to that using all training data. For the method with the self-supervised

first-stage training but without fine-tuning, although performance drops, the results using 10% training data are acceptable with the F_1 score of 47.5% and 43.8% on PANDA and JRDB-Group, respectively. Similarly, the full version of our method provides very promising results with only 10% training data on two benchmarks. We can see that, on PANDA benchmark with 10% training data, the full version ‘Ours’ produces 39.4% F_1 score, significantly outperforming the version without first-stage training (33.8% F_1 score). Similar results can be seen on the JRDB-Group benchmark. This demonstrates the effectiveness of the proposed self-supervised first-stage training strategy, which can reduce the dependence of the deep network on large-scale annotated training data. We can also see that ‘Ours w/o fine-tune.’ produces promising results when using a small amount of training data, e.g., 10% and 30%. This is because this setting without training the first-stage representation network significantly reduces the network parameters to be learned.

Table 4. Performance analysis using different amount of training data on PANDA and JRDB-Group datasets (%).

Method	Data	PANDA			JRDB-Group		
		Precision	Recall	F1	Precision	Recall	F1
Ours w/o 1 st -train	10% Train.	37.7	30.7	33.8	40.5	22.7	29.1
	30% Train.	41.9	34.7	38.0	39.1	46.0	42.2
	50% Train.	49.1	37.3	42.2	37.5	48.8	42.4
	100% Train.	48.4	41.3	44.6	49.8	47.8	48.8
Ours w/o fine-tune.	10% Train.	61.7	38.7	47.5	38.4	50.9	43.8
	30% Train.	54.1	44.0	48.5	51.6	44.0	47.5
	50% Train.	54.8	45.3	49.6	49.1	45.5	47.3
	100% Train.	58.6	45.3	51.1	50.3	48.1	49.2
Ours	10% Train.	43.5	36.0	39.4	42.2	46.4	44.2
	30% Train.	46.9	40.0	43.2	52.9	42.8	47.3
	50% Train.	55.1	50.7	52.8	53.9	42.4	47.4
	100% Train.	55.9	50.7	53.2	57.7	56.2	56.9

We further provide the group detection results using two new metrics. The commonly-used Half metric [6] takes a fixed threshold of 0.5 of group IOU for evaluating the group detection accuracy. In some cases, the threshold of 0.5 is a little loose, and a single threshold may bring in bias. This way, we vary the group IOU threshold of 0.5 in the classical Half metric to larger values until 1 and then calculate the corresponding group detection performance. We plot the curve using the abscissa axis as the threshold, and the vertical axis is the F_1 score under each threshold. Then we compute the AUC (Area Under The Curve) score as the metric for evaluation.

As a harsher metric, we also calculate the matrix IOU of the predicted group relation matrix $\hat{\mathbf{R}}$ and the ground-truth \mathbf{R}_{gt} by $\text{IOU}^{\text{GM}} = \frac{\sum \text{AND}(\hat{\mathbf{R}}, \mathbf{R}_{\text{gt}})}{\sum \text{OR}(\hat{\mathbf{R}}, \mathbf{R}_{\text{gt}})}$, where

AND and OR denotes the logical operations functions, and \sum denotes the summation over all values. The group relation metric encodes the group relations between each pair of subjects, and the metric IOU^{GM} evaluates the pair-wise group relation accuracy among all subjects.

Table 5. Comparative group detection results using IOU-AUC and IOU^{GM} (%).

	PANDA		JRDB-Group	
	IOU-AUC	IOU^{GM}	IOU-AUC	IOU^{GM}
Dis.Mat + [44]	17.0	5.9	14.1	12.9
GNN w GRU	16.3	6.5	21.7	20.1
ARG [40]	19.3	6.4	21.6	19.3
Joint [7]	-	-	20.4	16.6
JRDB-Group [9]	-	-	26.3	20.6
Ours	41.2	31.2	42.5	32.5

We show the results using the new metrics in Table 5. Note that, on the PANDA dataset, we do not conduct the experiments using the methods of Joint and JRDB-Group, since their deep CNN features can not be applied to the scene with thousands of people in the PANDA dataset. We can not provide the evaluation of new metrics for the baseline methods in PANDA, i.e., ‘Group-to-local’, because it did not provide the source code and raw results. From Table 5 we can first see that our method shows remarkable performance compared to the other methods. Specifically, on the JRDB dataset under the metrics of IOU-AUC and IOU^{GM} , the simple baseline method using the distance metric based method can not provide comparable or better results than other methods like that in Table 2. We can also see that the performance under IOU^{GM} is generally low. This also leaves space for the development of more effective algorithms.

5 Conclusion

In this paper, we have proposed a new framework for social group detection in large-scale multi-person scenes. We developed a double-head two-stage network for the human relation representation and group detection, in which we designed a self-supervised first-stage training strategy for relation representation. We evaluated the proposed method on two state-of-the-art benchmarks, i.e., PANDA and JRDB-Group. The proposed method outperforms all the comparison methods by a very large margin. We also verified the effectiveness of the self-supervised representation training strategy by using very few (labeled) training data.

Acknowledgment. This work was supported in part by the National Natural Science Foundation of China under Grants U1803264, 62072334, and the Tianjin Research Innovation Project for Postgraduate Students under Grant 2021YJSB174.

References

1. Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2d human pose estimation: New benchmark and state of the art analysis. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2014)
2. Bagautdinov, T., Alahi, A., Fleuret, F., Fua, P., Savarese, S.: Social scene understanding: End-to-end multi-person action localization and collective activity recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2017)
3. Bazzani, L., Cristani, M., Murino, V.: Decentralized particle filter for joint individual-group tracking. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2012)
4. Chang, M.C., Krahnstoeber, N., Ge, W.: Probabilistic group-level motion analysis and scenario recognition. In: *IEEE International Conference on Computer Vision* (2011)
5. Cheng, K., Zhang, Y., He, X., Chen, W., Cheng, J., Lu, H.: Skeleton-based action recognition with shift graph convolutional network. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020)
6. Choi, W., Chao, Y.W., Pantofaru, C., Savarese, S.: Discovering groups of people in images. In: *European Conference on Computer Vision* (2014)
7. Ehsanpour, M., Abedin, A., Saleh, F., Shi, J., Reid, I., Rezatofghi, H.: Joint learning of social groups, individuals action and sub-group activities in videos. In: *European Conference on Computer Vision* (2020)
8. Ehsanpour, M., Saleh, F., Savarese, S., Reid, I., Rezatofghi, H.: Jrdb-act: A large-scale multi-modal dataset for spatio-temporal action, social group and activity detection. *arXiv Preprint arXiv:2106.08827* (2021)
9. Ehsanpour, M., Saleh, F.S., Savarese, S., Reid, I.D., Rezatofghi, H.: Jrdb-act: A large-scale dataset for spatio-temporal action, social group and activity detection (2021)
10. Fan, L., Wang, W., Huang, S., Tang, X., Zhu, S.C.: Understanding human gaze communication by spatio-temporal graph reasoning. In: *IEEE/CVF International Conference on Computer Vision* (2019)
11. Feldmann, M., Fränken, D., Koch, W.: Tracking of extended objects and group targets using random matrices. *IEEE Transactions on Signal Processing* **59**(4), 1409–1420 (2010)
12. Fernando, T., Denman, S., Sridharan, S., Fookes, C.: Gd-gan: Generative adversarial networks for trajectory prediction and group detection in crowds. In: *Asian Conference on Computer Vision* (2018)
13. Gan, Y., Han, R., Yin, L., Feng, W., Wang, S.: Self-supervised multi-view multi-human association and tracking. In: *ACM International Conference on Multimedia* (2021)
14. Gavriluk, K., Sanford, R., Javan, M., Snoek, C.G.: Actor-transformers for group activity recognition. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020)
15. Ge, W., Collins, R.T., Ruback, R.B.: Vision-based analysis of small groups in pedestrian crowds. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **34**(5), 1003–1016 (2012)
16. Goel, A., Ma, K.T., Tan, C.: An end-to-end network for generating social relationship graphs. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019)

17. Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S., Alahi, A.: Social gan: Socially acceptable trajectories with generative adversarial networks. In: IEEE Conference on Computer Vision and Pattern Recognition (2018)
18. Han, R., Feng, W., Zhang, Y., Zhao, J., Wang, S.: Multiple human association and tracking from egocentric and complementary top views. IEEE TPAMI (2021). <https://doi.org/10.1109/TPAMI.2021.3070562>
19. Han, R., Feng, W., Zhao, J., Niu, Z., Zhang, Y., Wan, L., Wang, S.: Complementary-view multiple human tracking. In: AAAI Conference on Artificial Intelligence (2020)
20. Han, R., Gan, Y., Li, J., Wang, F., Feng, W., Wang, S.: Connecting the complementary-view videos: Joint camera identification and subject association. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2022)
21. Han, R., Wang, Y., Yan, H., Feng, W., Wang, S.: Multi-view multi-human association with deep assignment network. IEEE TIP **31**, 1830–1840 (2022)
22. Lerner, A., Chrysanthou, Y., Lischinski, D.: Crowds by example. In: Computer Graphics Forum (2007)
23. Li, W., Duan, Y., Lu, J., Feng, J., Zhou, J.: Graph-based social relation reasoning. In: European Conference on Computer Vision (2020)
24. Martín-Martín, R., Patel, M., Rezatofghi, H., Shenoi, A., Gwak, J., Frankel, E., Sadeghian, A., Savarese, S.: Jrdb: A dataset and benchmark of egocentric robot visual perception of humans in built environments. IEEE Transactions on Pattern Analysis and Machine Intelligence (2021)
25. Mohamed, A., Qian, K., Elhoseiny, M., Claudel, C.: Social-stgcn: A social spatio-temporal graph convolutional neural network for human trajectory prediction. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2020)
26. Monfort, M., Andonian, A., Zhou, B., Ramakrishnan, K., Bargal, S.A., Yan, T., Brown, L., Fan, Q., Gutfreund, D., Vondrick, C., et al.: Moments in time dataset: one million videos for event understanding. IEEE Transactions on Pattern Analysis and Machine Intelligence **42**(2), 502–508 (2019)
27. Moussaïd, M., Perozo, N., Garnier, S., Helbing, D., Theraulaz, G.: The walking behaviour of pedestrian social groups and its impact on crowd dynamics. PloS one **5**(4), e10047 (2010)
28. Pang, S.K., Li, J., Godsill, S.J.: Detection and tracking of coordinated groups. IEEE Transactions on Aerospace and Electronic Systems **47**(1), 472–502 (2011)
29. Pellegrini, S., Ess, A., Schindler, K., Van Gool, L.: You’ll never walk alone: Modeling social behavior for multi-target tracking. In: IEEE International Conference on Computer Vision (2009)
30. Pellegrini, S., Ess, A., Van Gool, L.: Improving data association by joint modeling of pedestrian trajectories and groupings. In: European Conference on Computer Vision (2010)
31. Pramono, R.R.A., Chen, Y.T., Fang, W.H.: Empowering relational network by self-attention augmented conditional random fields for group activity recognition. In: European Conference on Computer Vision (2020)
32. Shao, J., Dong, N., Zhao, Q.: A real-time algorithm for small group detection in medium density crowds. Pattern Recognition and Image Analysis **28**(2), 282–287 (2018)
33. Shao, J., Change Loy, C., Wang, X.: Scene-independent group profiling in crowd. In: IEEE Conference on Computer Vision and Pattern Recognition (2014)
34. Solera, F., Calderara, S., Cucchiara, R.: Socially constrained structural learning for groups detection in crowd. IEEE Transactions on Pattern Analysis and Machine Intelligence **38**(5), 995–1008 (2015)

35. Swofford, M., Peruzzi, J., Tsoi, N., Thompson, S., Martín-Martín, R., Savarese, S., Vázquez, M.: Improving social awareness through dante: Deep affinity network for clustering conversational interactants. *ACM on Human-Computer Interaction* 4(CSCW1), 1–23 (2020)
36. Thompson, S., Gupta, A., Gupta, A.W., Chen, A., Vázquez, M.: Conversational group detection with graph neural networks. In: *International Conference on Multimodal Interaction* (2021)
37. Turner, J.C.: Towards a cognitive redefinition of the social group. In: *Research Colloquium on Social Identity of the European Laboratory of Social Psychology*. Psychology Press (2010)
38. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: *Advances in Neural Information Processing Systems* (2017)
39. Wang, X., Zhang, X., Zhu, Y., Guo, Y., Yuan, X., Xiang, L., Wang, Z., Ding, G., Brady, D., Dai, Q., et al.: Panda: A gigapixel-level human-centric video dataset. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020)
40. Wu, J., Wang, L., Wang, L., Guo, J., Wu, G.: Learning actor relation graphs for group activity recognition. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019)
41. Yamaguchi, K., Berg, A.C., Ortiz, L.E., Berg, T.L.: Who are you with and where are you going? In: *IEEE Conference on Computer Vision and Pattern Recognition* (2011)
42. Yan, R., Xie, L., Tang, J., Shu, X., Tian, Q.: Social adaptive module for weakly-supervised group activity recognition. In: *European Conference on Computer Vision* (2020)
43. Yuan, H., Ni, D.: Learning visual context for group activity recognition. In: *AAAI Conference on Artificial Intelligence* (2021)
44. Zhan, X., Liu, Z., Yan, J., Lin, D., Loy, C.C.: Consensus-driven propagation in massive unlabeled data for face recognition. In: *European Conference on Computer Vision* (2018)
45. Zhao, J., Han, R., Gan, Y., Wan, L., Feng, W., Wang, S.: Human identification and interaction detection in cross-view multi-person videos with wearable cameras. In: *ACM International Conference on Multimedia* (2020)
46. Zhou, T., Wang, W., Qi, S., Ling, H., Shen, J.: Cascaded human-object interaction recognition. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020)
47. Zou, C., Wang, B., Hu, Y., Liu, J., Wu, Q., Zhao, Y., Li, B., Zhang, C., Zhang, C., Wei, Y., Sun, J.: End-to-end human object interaction detection with hoi transformer. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021)